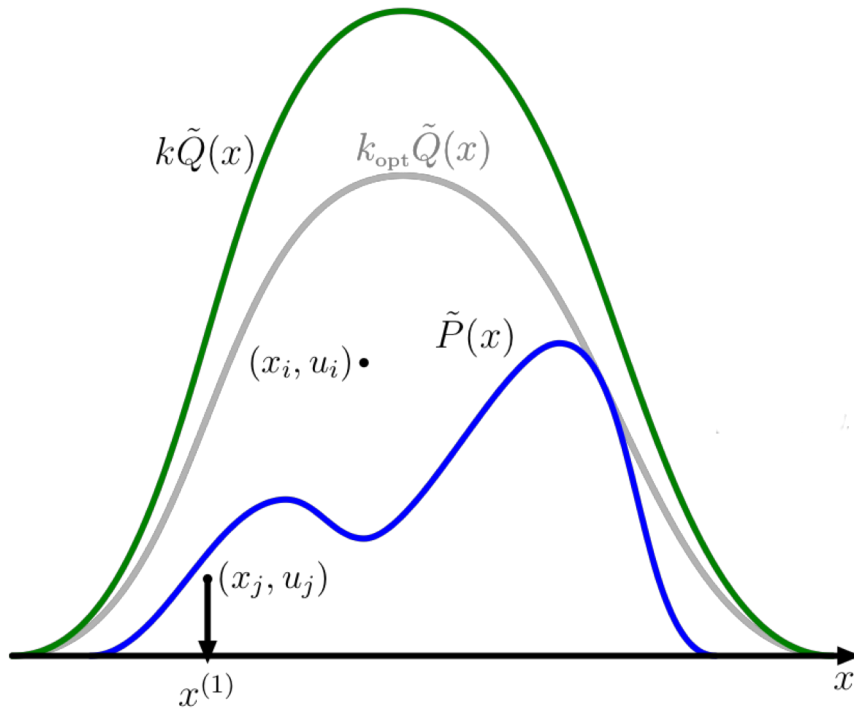


Slice Sampling and HMC

Kayhan Batmanghelich

Recap: Rejection Sampling



Steps:

- Find $Q(x)$ that is easy to sample from.
- Find k such that $kQ(x)$ is an upper bound for $\tilde{P}(x)$:

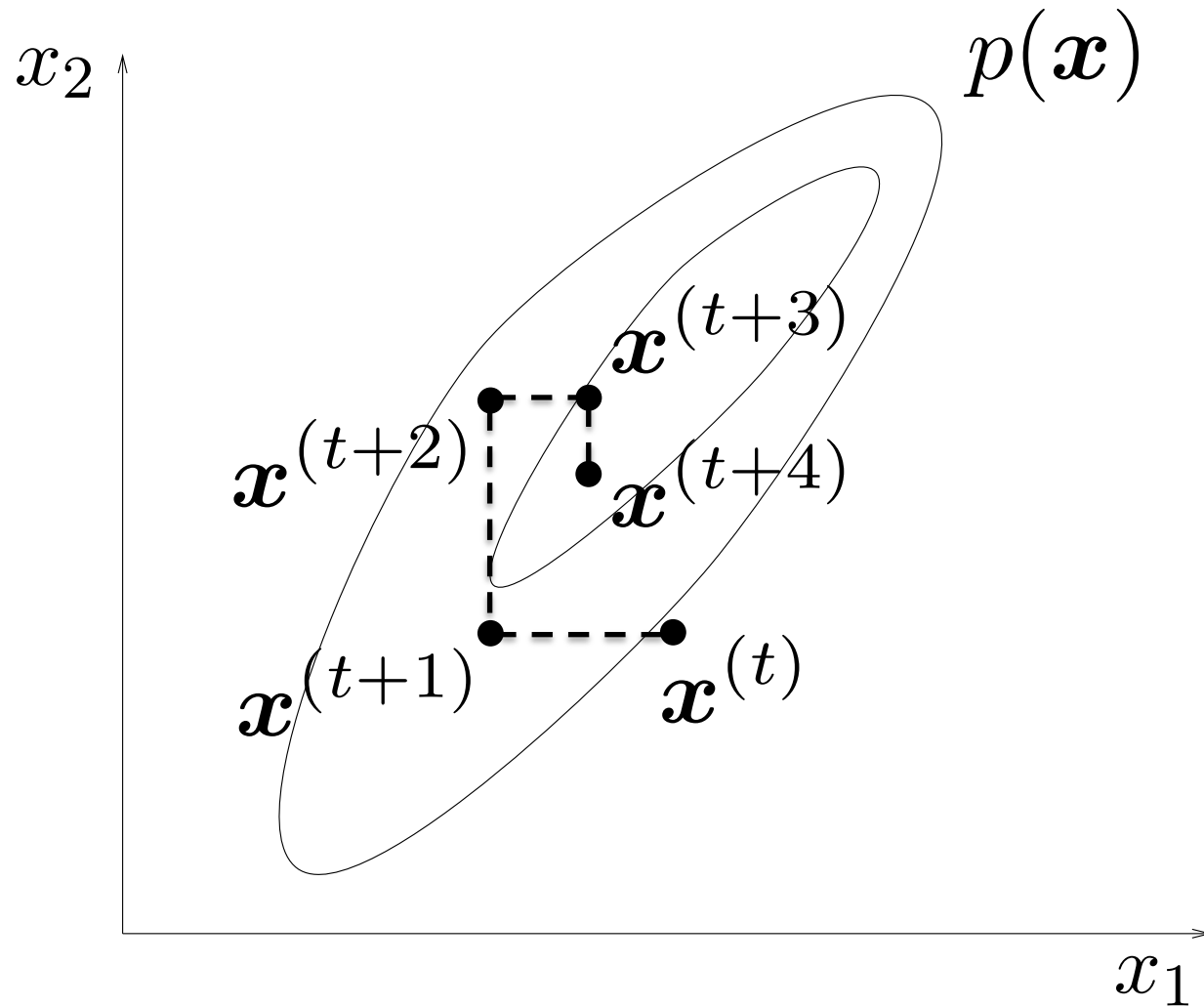
$$\frac{\tilde{P}(x)}{kQ(x)} < 1$$

- Sample auxiliary variable y

$$\mathbb{P}(y = 1|x) = \frac{\tilde{P}(x)}{kQ(x)}$$

accept the sample with probability $\mathbb{P}(y=1|x)$

Recap: Gibbs Sampling

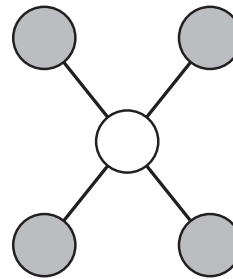


Ingredients for Gibb Recipe

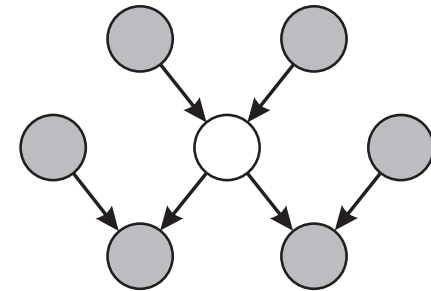
$$p(x_i | x_{\setminus i}) = \frac{1}{Z} p(x_i | \text{pa}(x_i)) \prod_{j \in \text{ch}(i)} p(x_j | \text{pa}(x_j))$$

Full conditionals only
need to condition on the
Markov Blanket

MRF

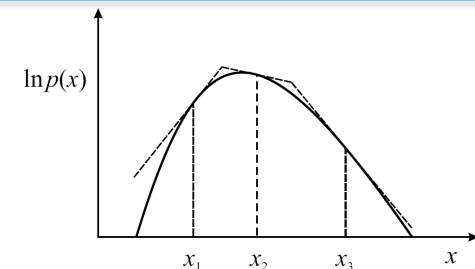


Bayes Net



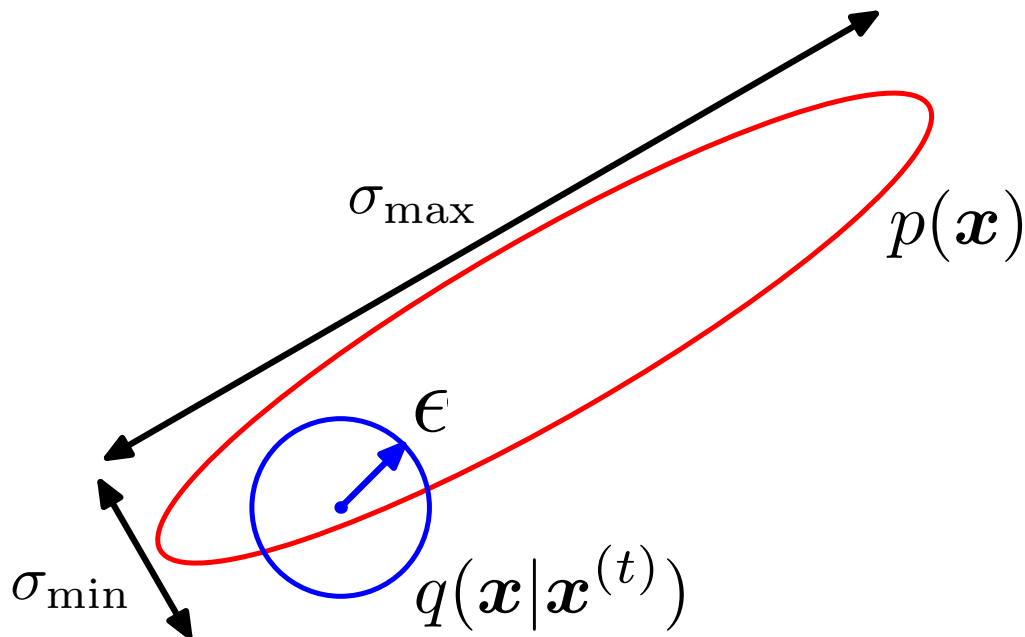
$$p(x_i | x_{\setminus i}) = \frac{1}{Z} p(x_i | \text{pa}(x_i)) \prod_{j \in \text{ch}(i)} p(x_j | \text{pa}(x_j))$$

- Must be “easy” to sample from conditionals
- Many conditionals are log-concave and are amenable to adaptive rejection sampling



Recap: Metropolis-Hastings

- For **Metropolis-Hastings**, a generic proposal distribution is: $q(x|x^{(t)}) = \mathcal{N}(0, \epsilon^2)$
- If ϵ is large, many rejections
- If ϵ is small, slow mixing

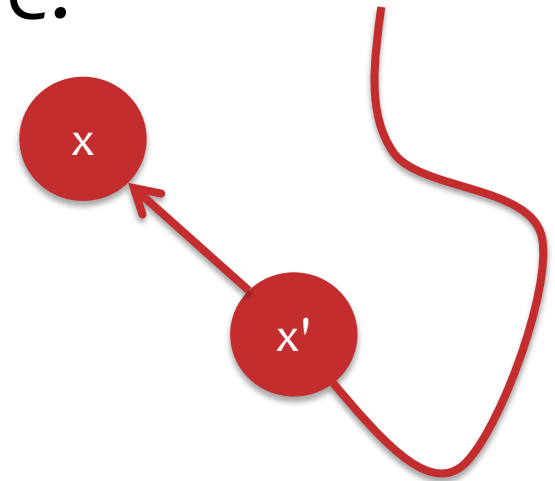
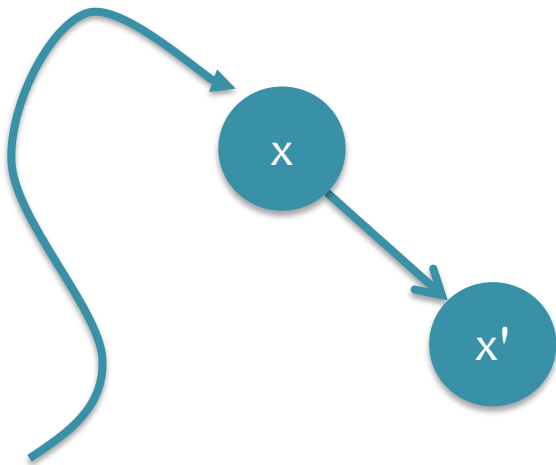


Recap: Detailed Balance

$$f(x', x) \tilde{q}(x' | x) p(x) = f(x, x') \tilde{q}(x | x') p(x')$$

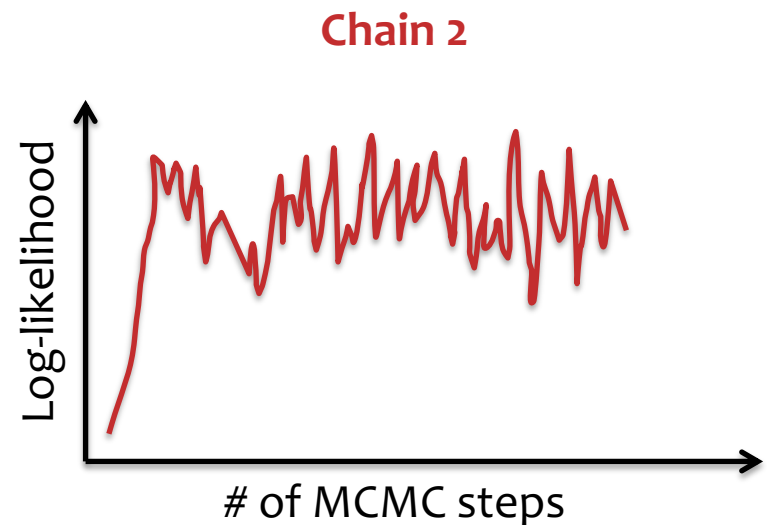
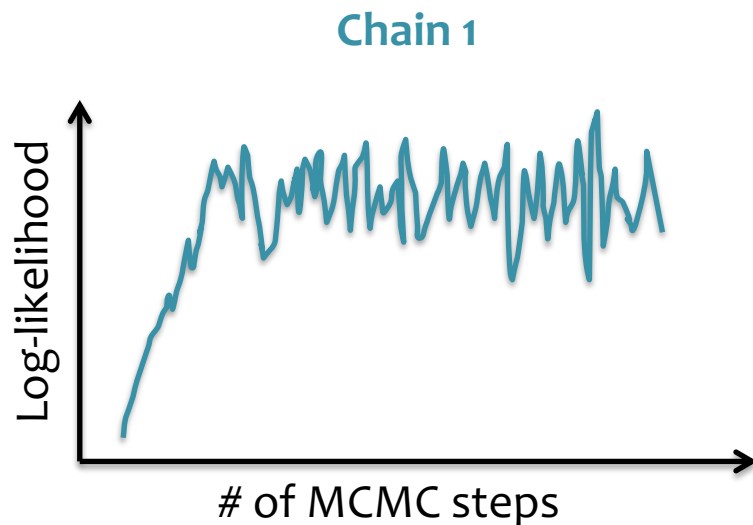
Detailed balance means that, for each pair of states x and x' ,

arriving at x then x' and arriving at x' then x are equiprobable.



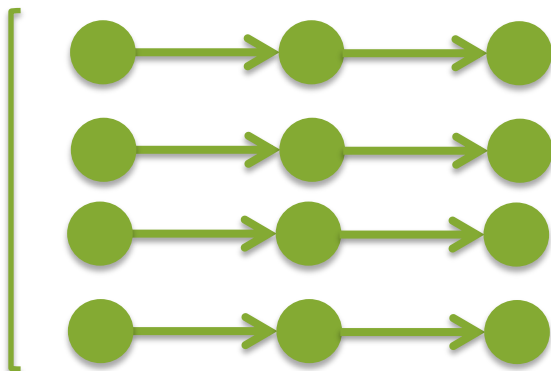
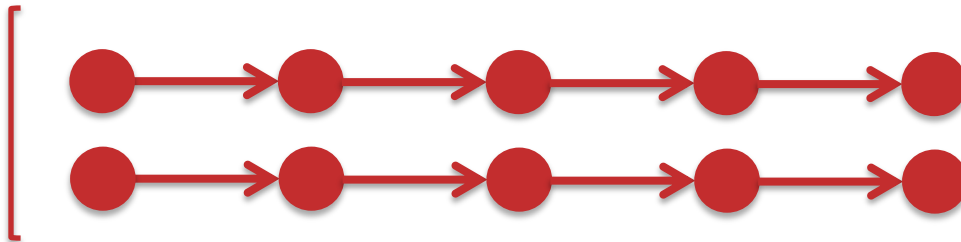
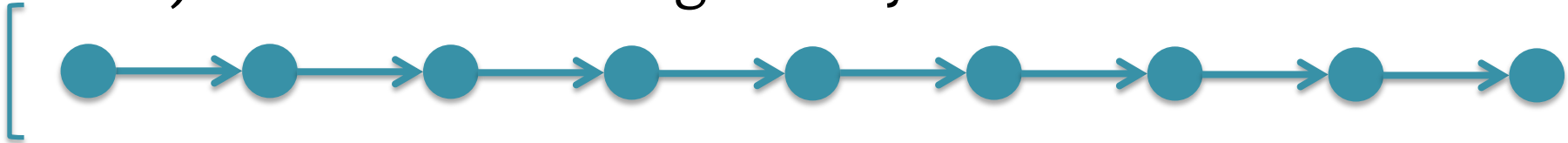
Recap: Practical Issues

- **Question:** How do we assess convergence of the Markov chain?
- **Answer:** It's not easy!
 - Compare statistics of multiple independent chains
 - Ex: Compare log-likelihoods



Recap: Practical Issues

- **Question:** Is one long Markov chain better than many short ones?
- **Note:** typical to discard initial samples (aka. “burn-in”) since the chain might not yet have mixed



- **Answer:** Often a balance is best:
 - Compared to one long chain: More independent samples
 - Compared to many small chains: Less samples discarded for burn-in
 - We can still parallelize
 - Allows us to assess mixing by comparing chains

Summary so far

General ideas for the sampling approaches

- Proposal distribution ($q(x)$): Use another distribution to sample from.
 - Change the proposal distribution with the iterations.
- Introduce an auxiliary variable to decide keeping a sample or not.
 - Why should we discard samples?
- Sampling from high-dimension is difficult.
 - Let's incorporate the graphical model into our sampling strategy.
- Can we use the gradient of the $p(x)$?

Slice Sampling, Hamiltonian Monte Carlo

MCMC (AUXILIARY VARIABLE METHODS)

Auxiliary variables

The point of MCMC is to marginalize out variables, but one can introduce more variables:

$$\begin{aligned}\int f(x)P(x) \, dx &= \int f(x)P(x, v) \, dx \, dv \\ &\approx \frac{1}{S} \sum_{s=1}^S f(x^{(s)}), \quad x, v \sim P(x, v)\end{aligned}$$

We might want to do this if

- $P(x|v)$ and $P(v|x)$ are simple
- $P(x, v)$ is otherwise easier to navigate

Auxiliary variables

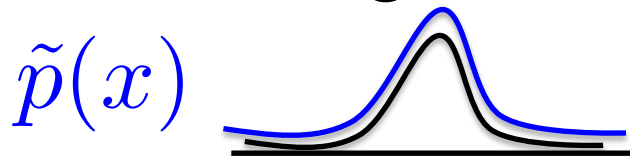
Consider drawing samples from $p(x)$. For an auxiliary variable y we introduce a distribution $p(y|x)$ to form the joint distribution:

$$p(x, y) = p(y|x)p(x)$$

- If we sampled x directly from $p(x)$ and then y from $p(y|x)$, introducing y is pointless!
- To be useful, therefore, the auxiliary variable must influence how we sample x .

Slice Sampling

- Motivation:
 - Want **samples** from $p(x)$ and don't know the normalizer Z
 - Choosing a proposal at the correct **scale** is difficult
- Properties:
 - Similar to *Gibbs Sampling*: **one-dimensional** transitions in the state space
 - Similar to *Rejection Sampling*: (asymptotically) draws samples from the **region under the curve**



- An MCMC method with an **adaptive proposal**

Slice sampling idea

By introducing the auxiliary variable y and defining the distribution

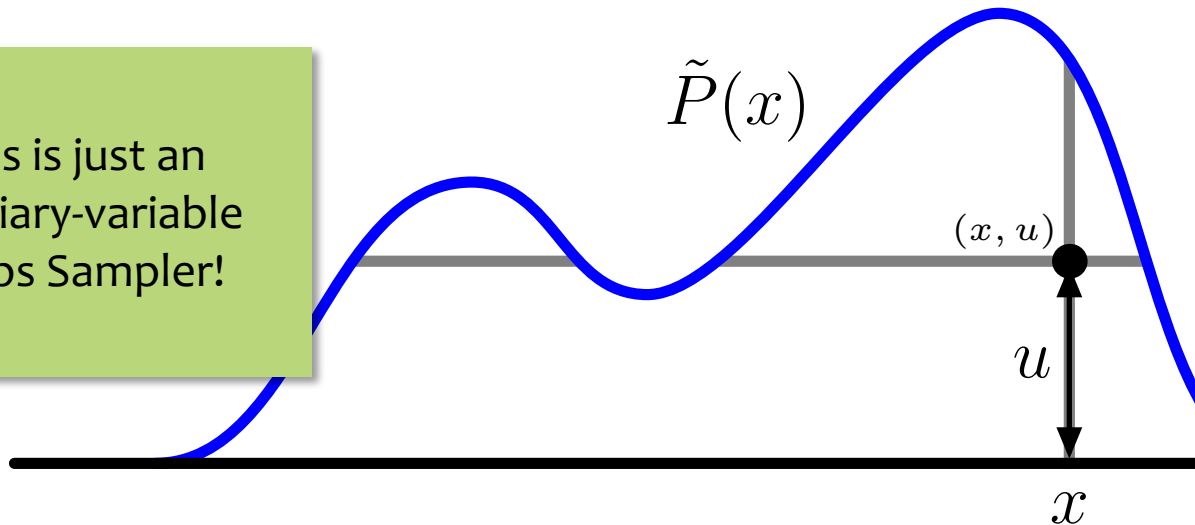
$$p(x, y) = \begin{cases} 1/Z & \text{for } 0 \leq y \leq p^*(x) \\ 0 & \text{otherwise} \end{cases}$$

White Board: Prove that the marginal of $p(x, y)$ over y is equal to the distribution we wish to draw samples from.

Slice sampling idea

Sample point uniformly under curve $\tilde{P}(x) \propto P(x)$

This is just an
auxiliary-variable
Gibbs Sampler!

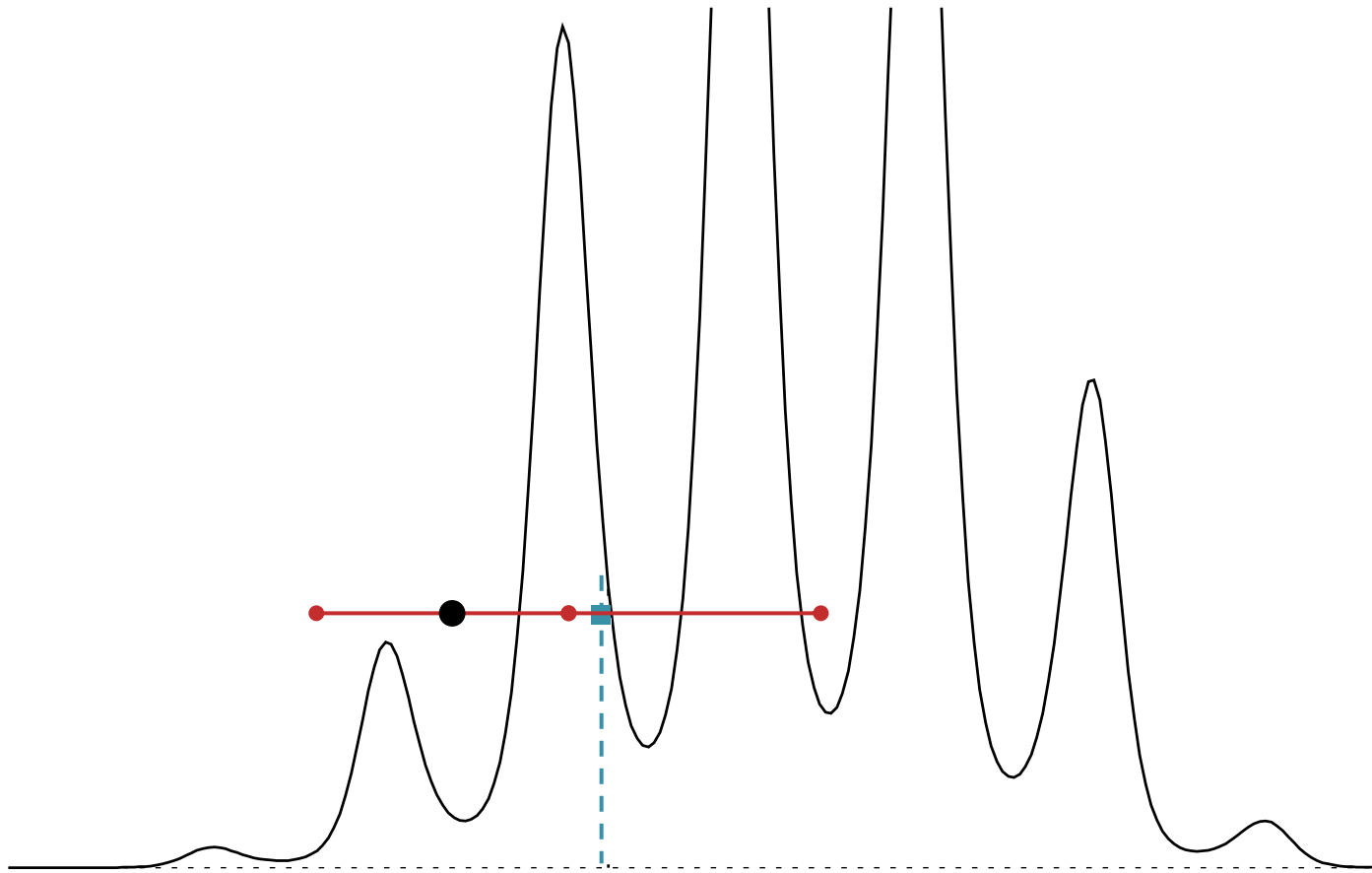


Problem: Sampling
from the conditional
 $p(x | u)$ might be
infeasible.

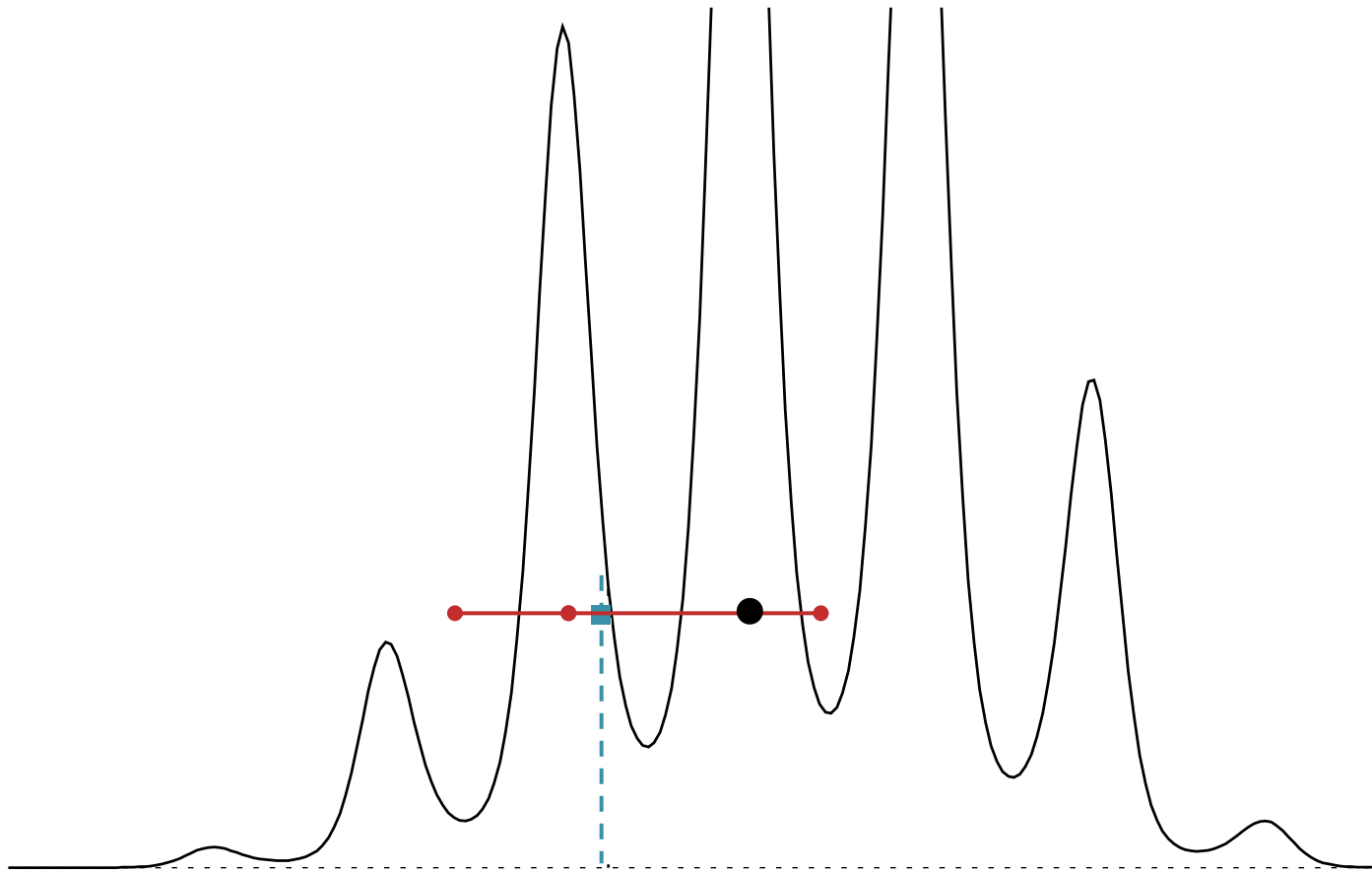
$$p(u|x) = \text{Uniform}[0, \tilde{P}(x)]$$

$$p(x|u) \propto \begin{cases} 1 & \tilde{P}(x) \geq u \\ 0 & \text{otherwise} \end{cases} = \text{"Uniform on the slice"}$$

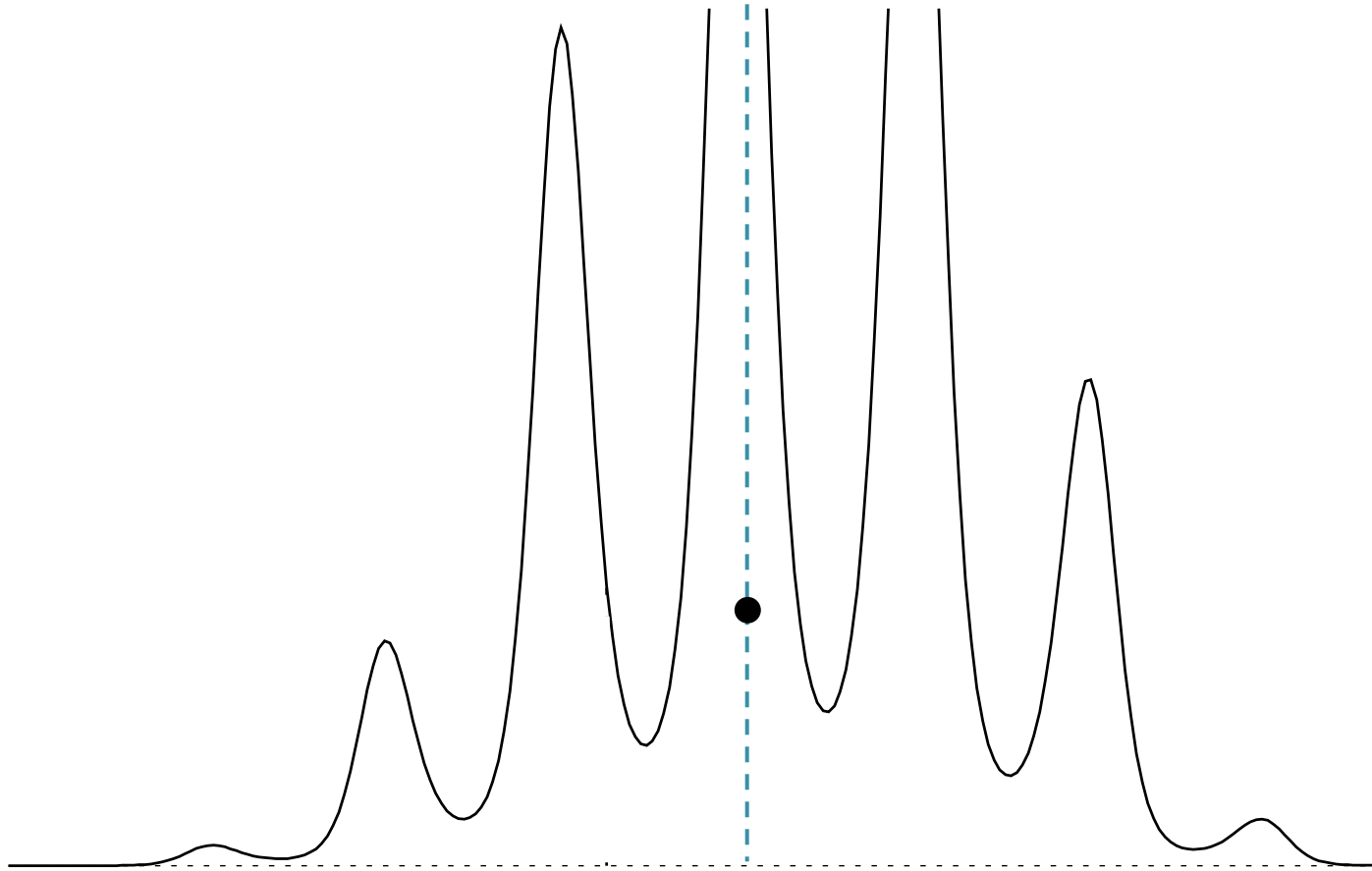
Slice Sampling



Slice Sampling



Slice Sampling



Slice Sampling

Goal: sample (x, u) given $(u^{(t)}, x^{(t)})$.

Part 1: Stepping Out

Sample interval (x_l, x_r) enclosing $x^{(t)}$.

Expand until endpoints are "outside" region under curve.

Part 2: Sample x (Shrinking)

Draw x from within the interval (x_l, x_r) , then accept or shrink.

Algorithm:

Slice Sampling

Goal: sample (x, u) given $(u^{(t)}, x^{(t)})$.

$u \sim \text{Uniform}(0, p(x^{(t)}))$

Part 1: Stepping Out

Sample interval (x_l, x_r) enclosing $x^{(t)}$.

$r \sim \text{Uniform}(u, w)$

$(x_l, x_r) = (x^{(t)} - r, x^{(t)} + w - r)$

Expand until endpoints are "outside" region under curve.

while($\tilde{p}(x_l) > u$) $\{x_l = x_l - w\}$

while($\tilde{p}(x_r) > u$) $\{x_r = x_r + w\}$

Part 2: Sample x (Shrinking)

Draw x from within the interval (x_l, x_r) , then accept or shrink.

Algorithm:

Slice Sampling

Goal: sample (x, u) given $(u^{(t)}, x^{(t)})$.

$u \sim \text{Uniform}(0, p(x^{(t)}))$

Part 1: Stepping Out

Sample interval (x_l, x_r) enclosing $x^{(t)}$.

$r \sim \text{Uniform}(u, w)$

$(x_l, x_r) = (x^{(t)} - r, x^{(t)} + w - r)$

Expand until endpoints are "outside" region under curve.

while($\tilde{p}(x_l) > u$) { $x_l = x_l - w$ }

while($\tilde{p}(x_r) > u$) { $x_r = x_r + w$ }

Part 2: Sample x (Shrinking)

while(true) {

Draw x from within the interval (x_l, x_r) , then accept or shrink.

$x \sim \text{Uniform}(x_l, x_r)$

if($\tilde{p}(x) > u$) { break }

else if($x > x^{(t)}$) { $x_r = x$ }

else { $x_l = x$ }

}

$x^{(t+1)} = x, u^{(t+1)} = u$

Algorithm:

Slice Sampling

Multivariate Distributions

- Resample each variable x_i **one-at-a-time** (just like Gibbs Sampling)

- Does not require sampling from

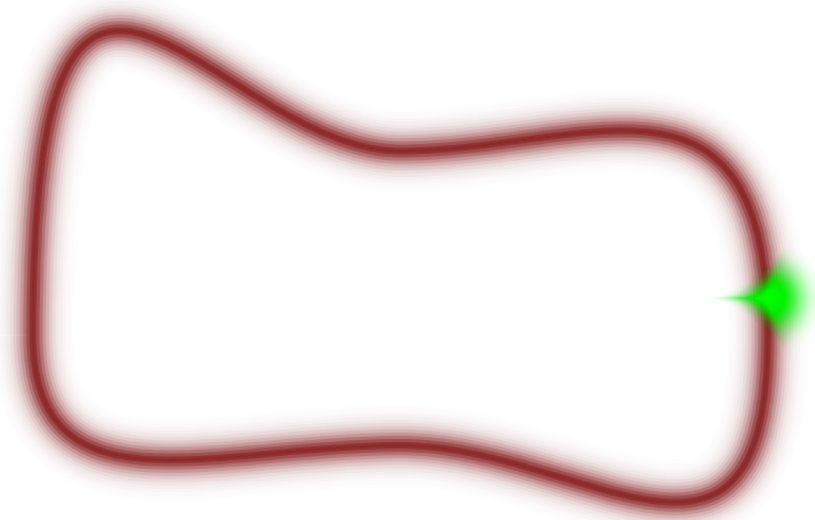
$$p(x_i | \{x_j\}_{j \neq i})$$

- Only need to evaluate a quantity **proportional** to the conditional

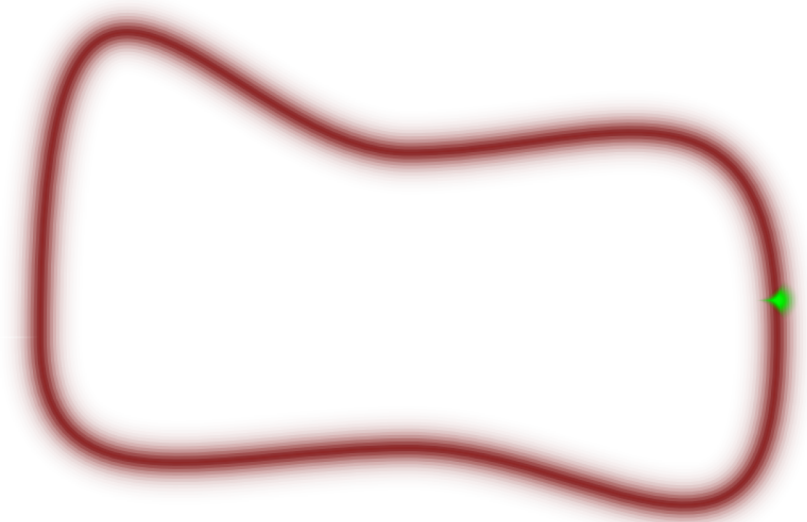
$$p(x_i | \{x_j\}_{j \neq i}) \propto \tilde{p}(x_i | \{x_j\}_{j \neq i})$$

Hamiltonian Monte Carlo

Example: Why MH is too slow?

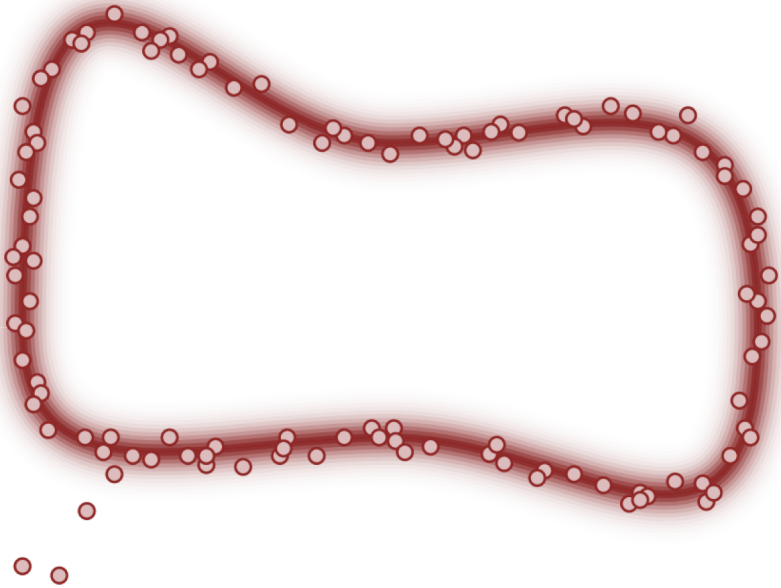


Large variance for the
proposal



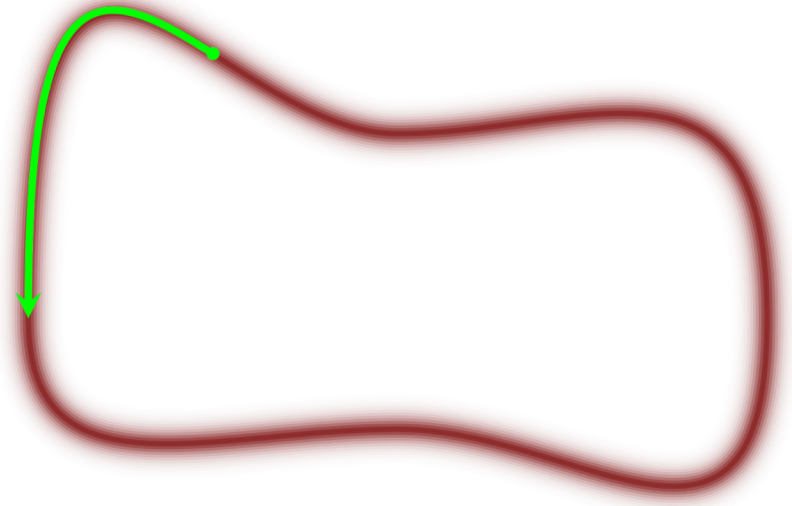
Small variance for the
proposal

Example: Why MH is too slow?



To get samples like this

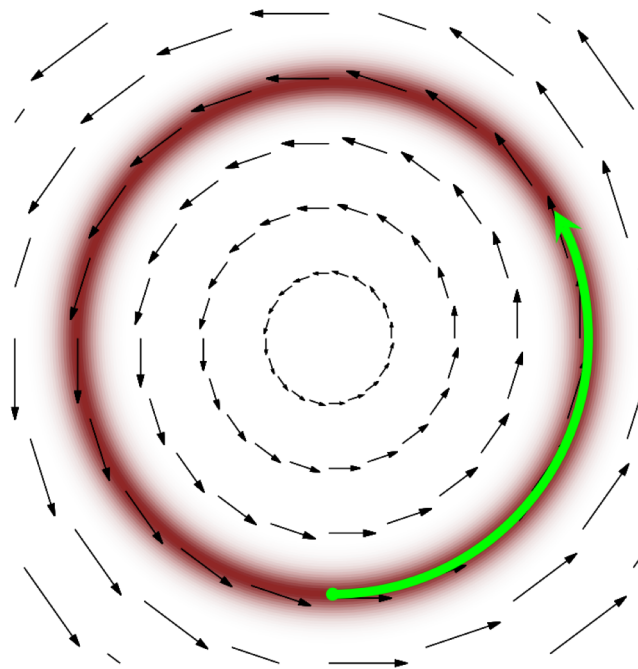
...



This how the trajectory
should look like!

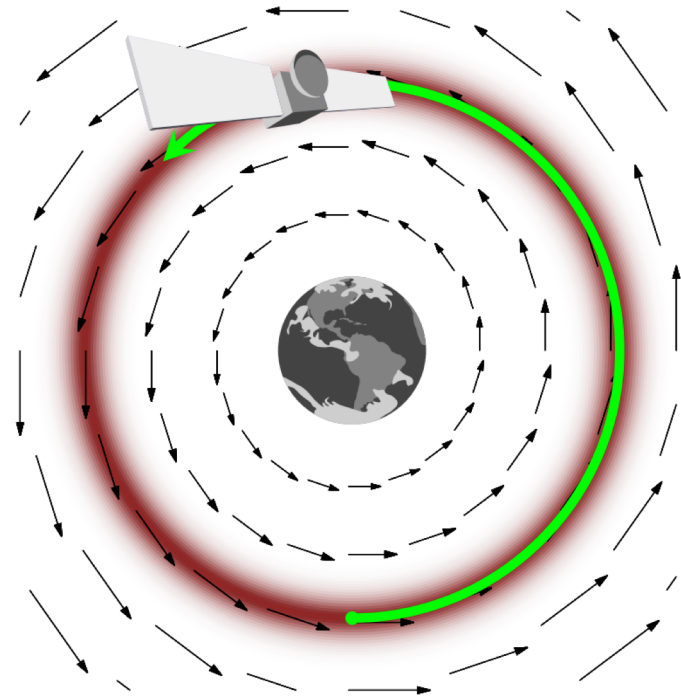
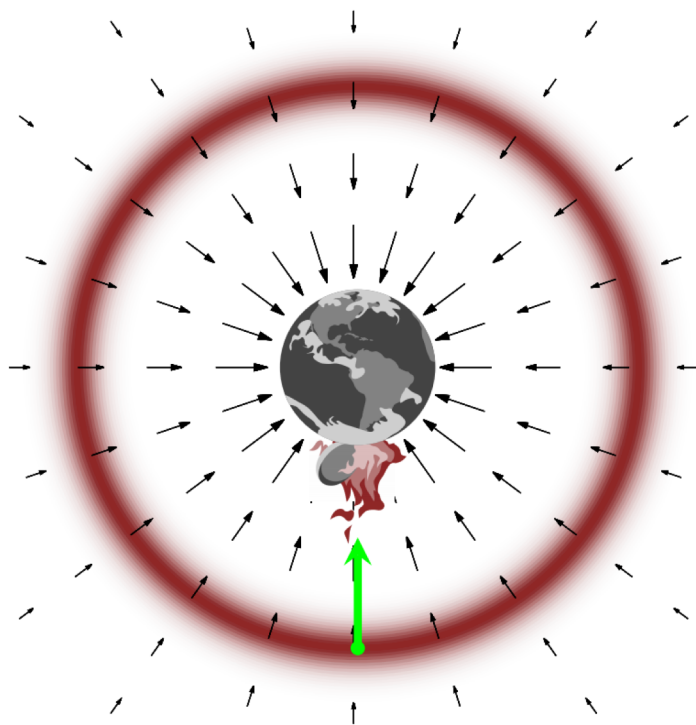
An Intuition from Physics

For every point in the parameter space, we need a vector field (assignment of a direction at every point) where the directions are aligned with the high probability regions.



How to come up with
the vector field?

An Intuition from Physics



the right amount of momentum to the physical system, the equations describing the evolution of the satellite define a vector field aligned with the orbit

Remember the Auxiliary Variable

Consider drawing samples from $\pi(x)$. For an auxiliary variable p we introduce a distribution $\pi(p|x)$ to form the joint distribution:

$$p(x, y) = p(y|x)p(x) \tau(x)$$

I use $\pi(\cdot, \cdot)$ to denote the probability. I want to use p for a different thing!

The latent variables

The auxiliary variable (moment)
 $\dim(x) = \dim(p)$

Remember the Auxiliary Variable

The expanded system defines a *Hamiltonian* that decomposes into a *potential* energy and *kinematic* energy.

$$\pi(x, p) = \pi(p|x)\pi(x)$$

$$\begin{aligned} H(x, p) &= -\log \pi(x, p) \\ &= \underbrace{-\log \pi(p|x)}_{\substack{\text{Kinematic Energy} \\ K(p,x)}} - \underbrace{\log \pi(x)}_{\substack{\text{Potential Energy} \\ E(x)}} \end{aligned}$$

Hamiltonian Monte Carlo

- Suppose we have a distribution of the form:

$$\pi(x) = \exp\{-E(x)\}/Z$$

where $x \in \mathcal{R}^N$

- We could use **random-walk M-H** to draw samples, but it seems a shame to **discard gradient information** $\nabla_x E(x)$
- If we can evaluate it, the gradient tells us where to look for **high-probability regions!**


Background: Hamiltonian Dynamics

Applications:

- Following the motion of atoms in a fluid through time
- Integrating the motion of a solar system over time
- Considering the evolution of a galaxy (i.e. the motion of its stars)
- “molecular dynamics”
- “N-body simulations”

Properties:

- Total energy of the system $H(x,p)$ stays constant
- Dynamics are reversible



Important for
detailed balance

Hamiltonian Dynamic

This is how we get the vector field:

$$\frac{dx}{dt} = \frac{\partial K(x, p)}{\partial p}$$

Acts like a velocity

$$\frac{dp}{dt} = -\frac{\partial K(x, p)}{\partial x} - \frac{\partial E(x)}{\partial x}$$

Acts like a correction for the gradient Gradient of the energy

A Choice for Kinematics

Let $\boldsymbol{x} \in \mathcal{R}^N$ be a position

$\boldsymbol{p} \in \mathcal{R}^N$ be a momentum

Potential energy: $E(\boldsymbol{x})$

Kinetic energy: $K(\boldsymbol{p}) = \boldsymbol{p}^T \boldsymbol{p} / 2$

Total energy: $H(\boldsymbol{x}, \boldsymbol{p}) = E(\boldsymbol{x}) + K(\boldsymbol{p})$



Hamiltonian function

A Choice for Kinematics

Let $\boldsymbol{x} \in \mathcal{R}^N$ be a position

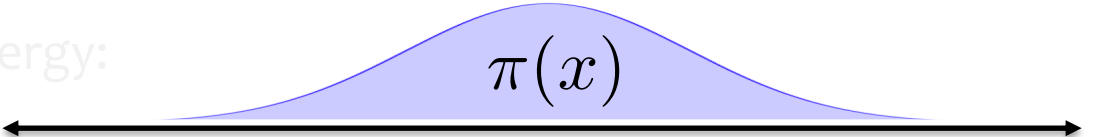
$\boldsymbol{p} \in \mathcal{R}^N$ be a momentum

Potential energy: $E(\boldsymbol{x})$

Kinetic energy:

Total energy:

$$H(\boldsymbol{x}, \boldsymbol{p}) = E(\boldsymbol{x}) + K(\boldsymbol{p})$$



$\pi(\boldsymbol{x})$

\boldsymbol{x}

A Choice for Kinematics

Let $\mathbf{x} \in \mathcal{R}^N$ be a position

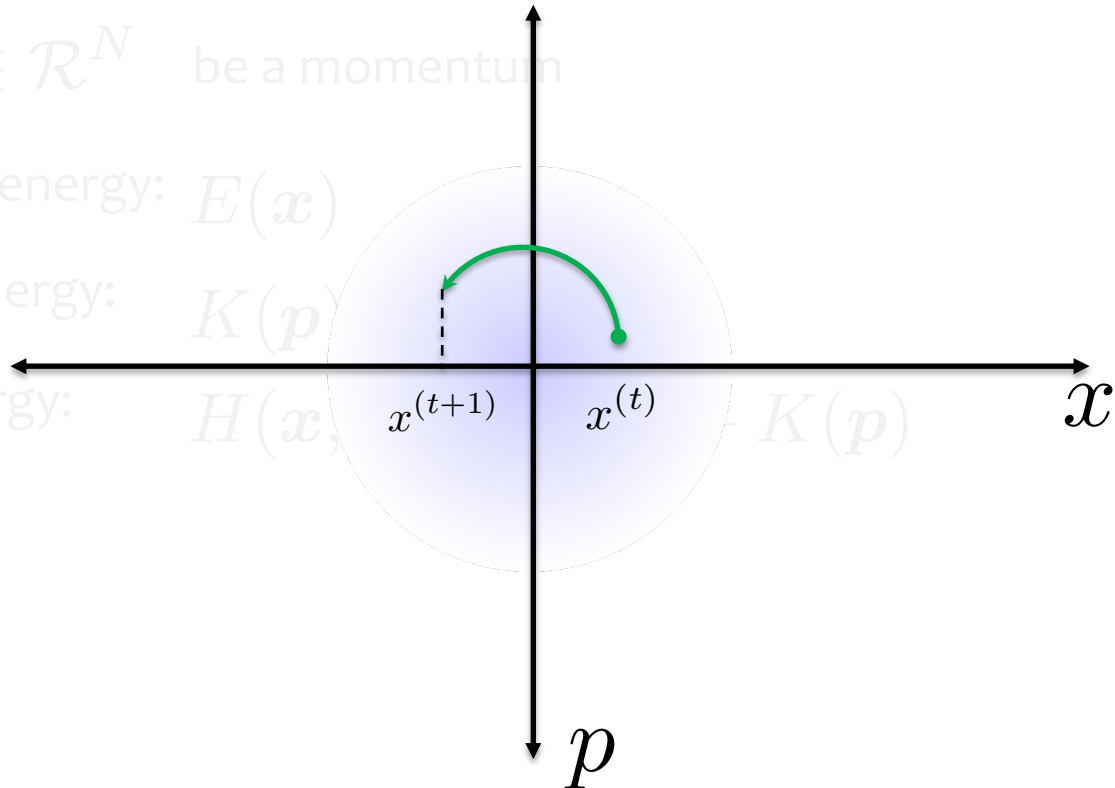
$\mathbf{p} \in \mathcal{R}^N$ be a momentum

Potential energy: $E(\mathbf{x})$

Kinetic energy: $K(\mathbf{p})$

Total energy:

$H(\mathbf{x}, \mathbf{p}) = E(\mathbf{x}) + K(\mathbf{p})$



A Choice for Kinematics

Let $\boldsymbol{x} \in \mathcal{R}^N$ be a position

$\boldsymbol{p} \in \mathcal{R}^N$ be a momentum

Potential energy: $E(\boldsymbol{x})$

Kinetic energy: $K(\boldsymbol{p}) = \boldsymbol{p}^T \boldsymbol{p} / 2$

Total energy: $H(\boldsymbol{x}, \boldsymbol{p}) = E(\boldsymbol{x}) + K(\boldsymbol{p})$



Hamiltonian function

How to simulate the dynamic:

Given a starting position $\boldsymbol{x}^{(l)}$ and a starting momentum $\boldsymbol{p}^{(l)}$ we can simulate the Hamiltonian dynamics of the system via:

1. Euler's method
2. Leapfrog method
3. etc.

Background: Hamiltonian Dynamics

Parameters to tune:

1. Step size, ϵ
2. Number of iterations, L

Leapfrog Algorithm:

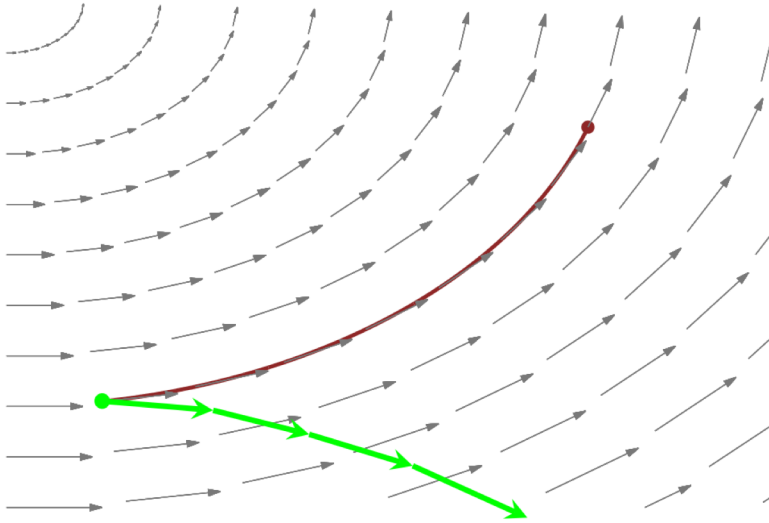
for τ in $1 \dots L$:

$$\mathbf{p} = \mathbf{p} - \frac{\epsilon}{2} \nabla_{\mathbf{x}} E(\mathbf{x})$$

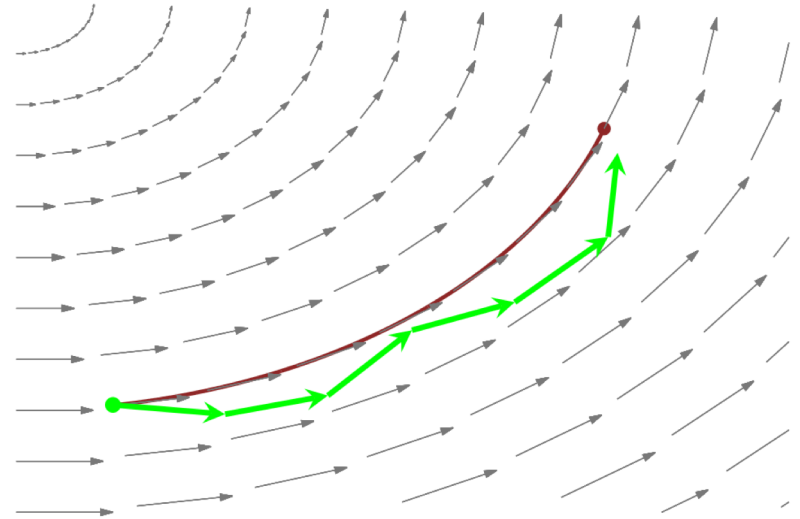
$$\mathbf{x} = \mathbf{x} + \epsilon \mathbf{p}$$

$$\mathbf{p} = \mathbf{p} - \frac{\epsilon}{2} \nabla_{\mathbf{x}} E(\mathbf{x})$$

Different Integration Schemes



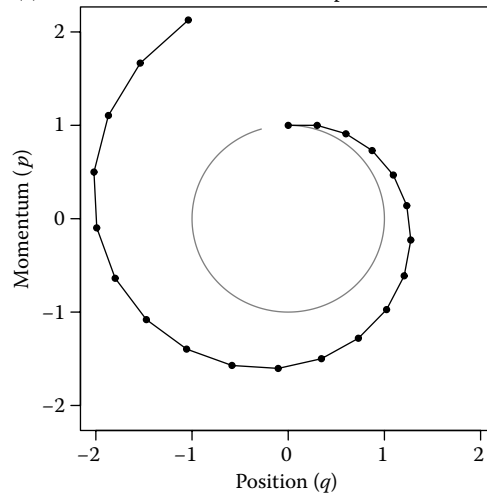
Most numerical integrators tend to drift away. As the system is integrated longer and longer, errors add coherently and push the numerical trajectory away from the exact trajectory.



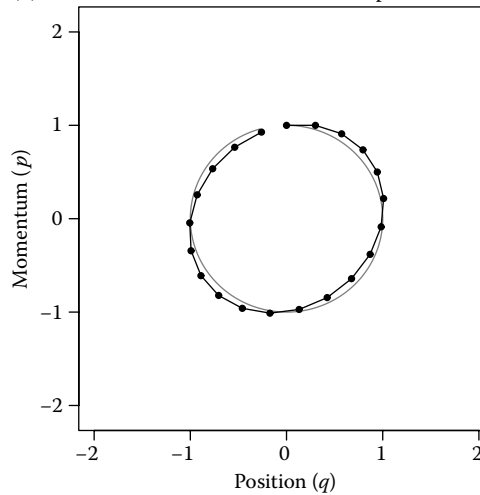
Leaf frog: the numerical trajectories oscillate around the exact level set, even as we integrate for longer and longer times.

Different Integration Schemes

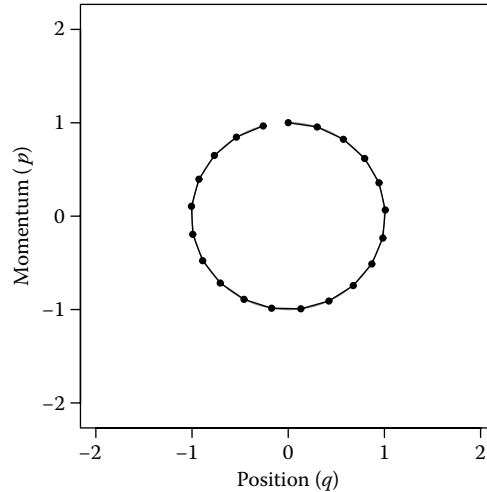
(a) Euler's method, stepsize 0.3



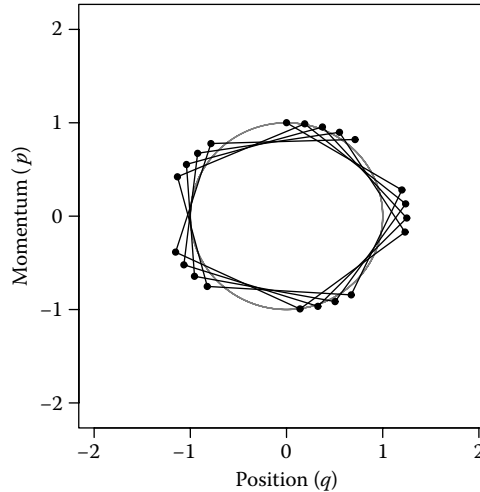
(b) Modified Euler's method, stepsize 0.3



(c) Leapfrog method, stepsize 0.3



(d) Leapfrog method, stepsize 1.2



Hamiltonian Monte Carlo

Preliminaries

Goal: $p(\mathbf{x}) = \exp\{-E(\mathbf{x})\}/Z$ where $\mathbf{x} \in \mathcal{R}^N$

Define: $K(\mathbf{p}) = \mathbf{p}^T \mathbf{p} / 2$
 $H(\mathbf{x}, \mathbf{p}) = E(\mathbf{x}) + K(\mathbf{p})$
 $p(\mathbf{x}, \mathbf{p}) = \exp\{-H(\mathbf{x}, \mathbf{p})\} / Z_H$
 $= \exp\{-E(\mathbf{x})\} \exp\{-K(\mathbf{p})\} / Z_H$

Note:

Since $p(\mathbf{x}, \mathbf{p})$ is
separable...

$$\Rightarrow \sum_{\mathbf{p}} p(\mathbf{x}, \mathbf{p}) = \exp\{-E(\mathbf{x})\} / Z$$

Target dist.

$$\Rightarrow \sum_{\mathbf{x}} p(\mathbf{x}, \mathbf{p}) = \exp\{-K(\mathbf{p})\} / Z_K$$

Gaussian

HMC Algorithm

1. Sample momentum (p) from distribution implied by the kinetic $\pi(p|x)$.
2. Update (x, p) according to Hamiltonian Dynamics

$$x \leftarrow x + \epsilon \frac{\partial K}{\partial p}$$
$$p \leftarrow p - \epsilon \left(\frac{\partial K}{\partial x} + \frac{\partial E}{\partial x} \right)$$

3. Accept/Reject the new sample

$$\pi(\text{accept}) = \min \left(1, \frac{\pi(\Phi_\tau(x, p))}{\pi(x, p)} \right)$$

HMC Algorithm

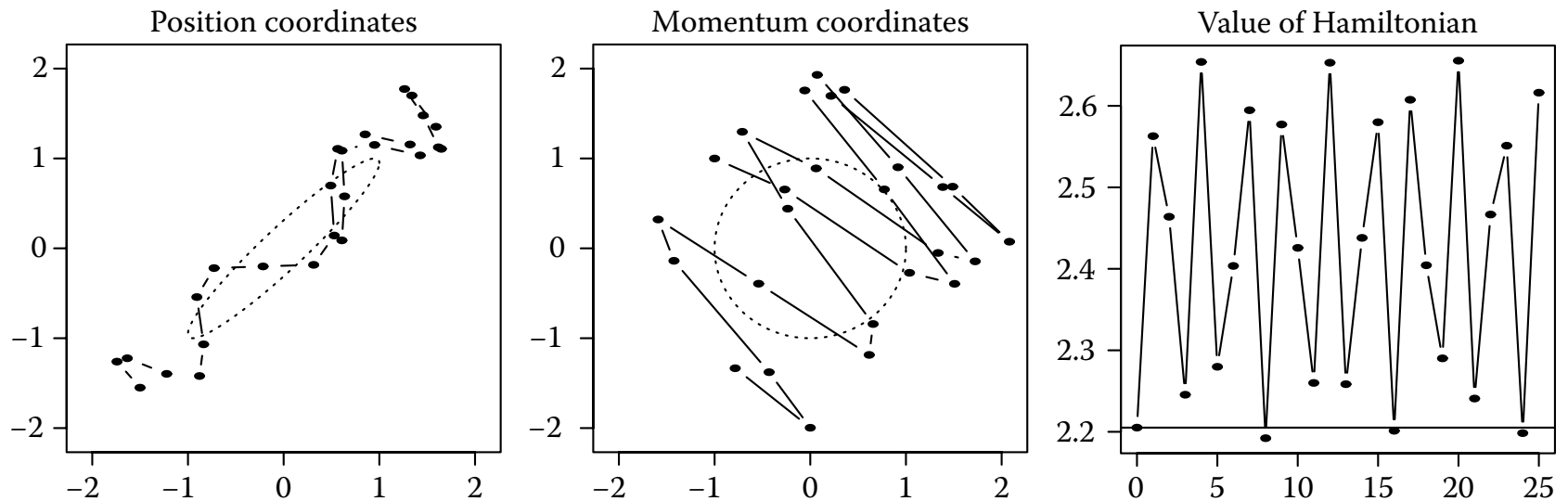
1. Sample momentum (p) from distribution implied by the kinetic $\pi(p|x)$.
2. Update (x, p) according to Hamiltonian Dynamics

$$x \leftarrow x + \epsilon \frac{\partial K}{\partial p}$$
$$p \leftarrow p - \epsilon \left(\frac{\partial K}{\partial x} + \frac{\partial E}{\partial x} \right)$$

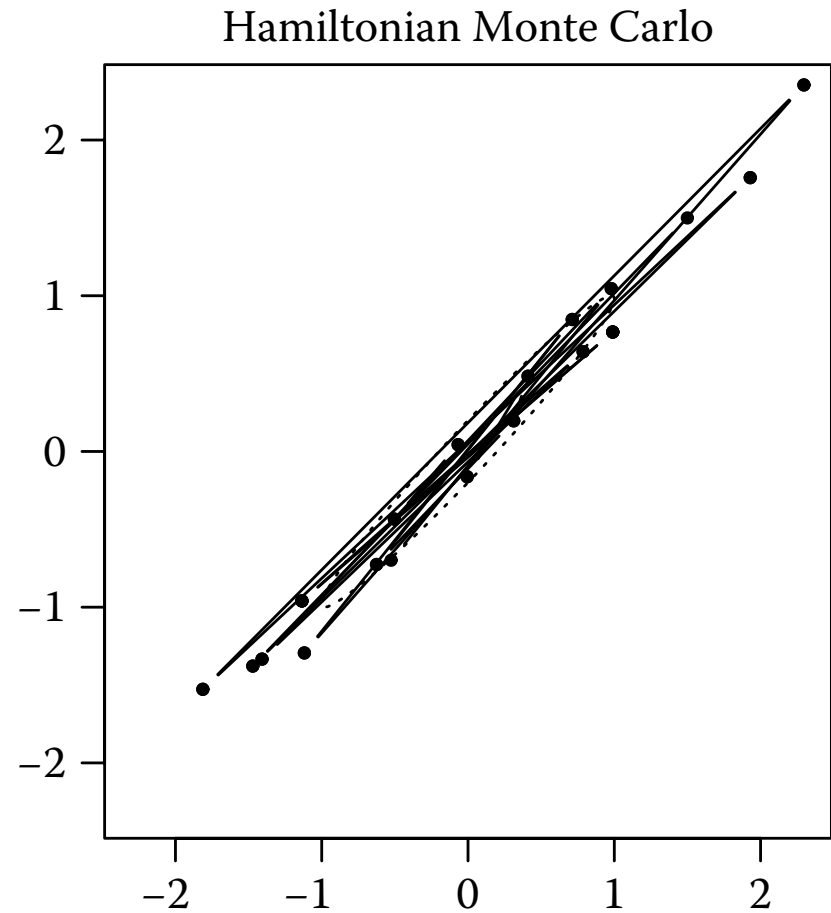
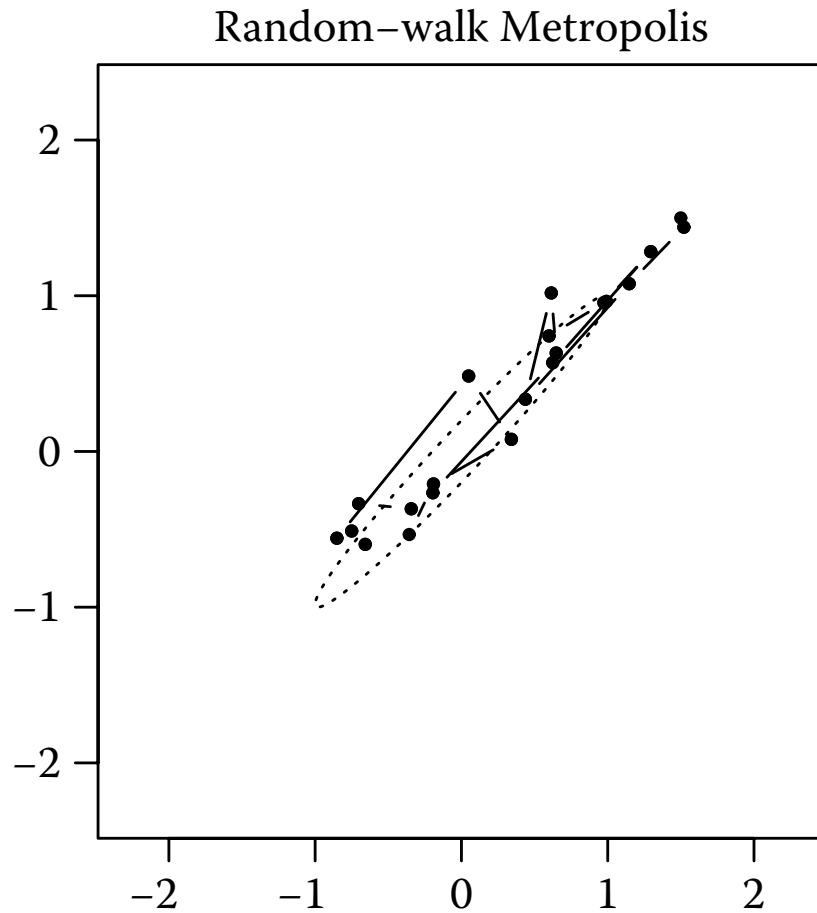
3. Accept/Reject the new sample

$$\pi(\text{accept}) = \min \left(1, \frac{\pi(\Phi_\tau(x, p))}{\pi(x, p)} \right)$$

Hamiltonian Monte Carlo



M-H vs. HMC



Simulations of MCMC

Visualization of Metropolis-Hastings, Gibbs Sampling, and Hamiltonian MCMC:

<http://twiecki.github.io/blog/2014/01/02/visualizing-mcmc/>

HMC in 2018

ABOUT USERS DEVELOPERS EVENTS SHOP SUPPORT



Stan

<http://mc-stan.org/>

The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo

Matthew D. Hoffman

*Adobe Research
601 Townsend St.
San Francisco, CA 94110, USA*

MATHOFFM@ADOBE.COM

Andrew Gelman

*Departments of Statistics and Political Science
Columbia University
New York, NY 10027, USA*

GELMAN@STAT.COLUMBIA.EDU

MCMC Summary

- **Pros**

- Very general purpose
- Often easy to implement
- Good theoretical guarantees as $t \rightarrow \infty$

- **Cons**

- Lots of tunable parameters / design choices
- Can be quite slow to converge
- Difficult to tell whether it's working